REMARKS

Claim 5 has been amended to correct a minor dependency issue. Claims 1-20 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Section 103(a) Rejections:

The Examiner rejected claims 1, 12 and 20 under 35 U.S.C. § 103(a) as being unpatentable over Aldred et al. (U.S. Patent 5,719,942) (hereinafter "Aldred") in view of Raynak et al. (U.S. Patent 5,680,549) (hereinafter "Raynak"), claims 2-6, 8-11, 13-17 and 19 as being unpatentable over Aldred and Raynak, and in further view of Simonoff et al. (U.S. Patent 6,005,568) (hereinafter "Simonoff"), and claims 17 and 18 as being unpatentable over Aldred, Raynak and Simonoff, and in further view of Jalili et al. (U.S. Patent 5,423,042) (hereinafter "Jalili"). Applicant respectfully traverses these rejections for at least the reasons provided below.

Regarding claim 1, contrary to the Examiner's assertion, Aldred in view of the Raynak does not teach or suggest a first application launching a second application, where the launching of the second application includes the first application passing an event port number and a command port number to the second application. Aldred specifically teaches the use of support system software together with call manager applications to establish, configure, and manage communication channels between applications, especially between applications executing on different hardware nodes (Abstract; column 1, lines 52-60). Aldred teaches that groups of applications communicate by participating in named sharing sets. Aldred's call managers coordinate, monitor and manage the various share sets of applications. Aldred also teaches a support system and a software API through which applications interact with the call managers. Aldred's API includes functions for initiating and configuring communication between shared applications via channels and signals.

The Examiner cites various portions of Aldred (column 5, lines 51-63; column 6, lines 39-49; column 7, lines 33-62; column 12, lines 57-61; and column 36, lines 3-52) that describe Aldred's channels and share sets. **However, none of these cited passages describes passing event port numbers and command port numbers to an application as part of launching that application.** Instead, Aldred teaches a manner and method of initializing and configuring communication channels between applications that does not include passing event port numbers and command port numbers as part of launching an application. Aldred explains the benefits of using the support system software when establishing and configuring communications channels. For example, relying upon call managers and the support system software allows applications to be "aware" of, and to use, Aldred's system while avoiding the need to be involved in "call set-up or tear-down." Aldred teaches the benefit of providing clear separation of call management and application programming (Aldred, column 26, lines 62-67).

Aldred states, "in order for an application instance to be allowed to communicate with the system, it must identify itself by issuing a register_app call" (column 35, lines 48-67). Aldred also teaches, "it is up to the launched application to use [the] register_app [function] to fully identify itself to the system" (column 36, lines 21-55). Aldred describes that adding a port to a channel includes a request from one application, which is sent via the support system as an unsolicited event to a second application, and a confirmation (or error) response routed back to the first application as a confirm event. (Aldred, column 24, lines 39-51). Additionally, one of the benefits of Aldred's share sets, call managers and support system software is that data may be communicated across heterogeneous networks using passive nodes to route data between an application on one node and another application on another node (Aldred, column 2, lines 19-50; column 5, lines 41-50; column 19, lines 24-48). Nowhere does Aldred describe passing event port numbers and command port numbers to an application *as part of launching that application*.

The Examiner also cites column 11, lines 27-39 and column 29, lines 8-19, where Aldred describes launching applications and refers to Aldred's "launch_app" API command. However, Aldred's launch_app function is used by applications to interact with, and request support services from, Aldred's call managers (*see*, Aldred, column 4, lines 27-43). Thus, an application wishing to launch another application uses the launch_app function to communicate the request to a call manager. The call manager forwards the request to a call manager executing on the appropriate node of Aldred's system. The second call manager may then launch the requested application (Aldred, column 5, lines 51-63). The fact that Aldred includes a mechanism to launch applications does not teach or suggest passing an event port number and a command port number as part of launching an application. Nowhere does Aldred describe passing event port numbers and command port numbers as part of launching an application via the launch_app API function. In contrast, Aldred teaches that an application may issue a launch_app command and may be returned a limited use handle to the launched application that is "only valid in very restricted circumstances *until the launched application has registered with the support system*" (emphasis added, column 11, lines 34-36). Thus, as noted above, Aldred teaches that ports, and therefore event port numbers and command port numbers, are only configured after an application has registered with the support system.

The Examiner, in the Response to Arguments section, "believes it is reasonable to suggest that the parameters passed in Aldred's launch_app function would be the channel characteristics needed for launching and launched applications to communicate". The Examiner's belief is completely unsupported by the teachings of the cited art, and can thus only be based on hindsight knowledge of Applicants' disclosure. In fact, Aldred actually teaches away from one application passing event port numbers and command port numbers as part of launching another application. As described above, Aldred's system already includes a very specific mechanism to initiate and configure ports and channels between applications that specifically does not include passing event port numbers and command port numbers as part of launching applications. Aldred clearly teaches the benefits of an application first registering with a call manager and joining a share set before initiating or configuring channels and ports. Rather than providing any motivation to modify Aldred's system to pass event port numbers and command port numbers as part of one application launching another application, **Aldred teaches away** from one application launching another application and passing event port numbers and command port numbers as part of launching the other application. Furthermore, it would not make sense to modify Aldred to bypass the share sets that are central to Aldred's system by passing event port numbers and command port numbers as part of launching applications. Such a modification would not only be contrary to Aldred's specific teachings, it would remove the specific benefits of Aldred's sharing sets, call managers, and support system software.

The Examiner also asserts that, "the sending application is responsible for defining the channel characteristics", and that "modification of Aldred to include passing an channel and port information between applications does not change the principle of his invention". However, the Examiner has clearly mischaracterized Aldred's meaning in "the sending application is responsible for defining the channel characteristics" by stating instead "the sending application is responsible for establishing the channel between applications". *Defining the channel characteristics* and *establishing the channel* are two different actions. In fact, Aldred describes these characteristics in 12:57-64 as: "target application handle, channel set type and identifier, data class, maximum buffer size, user exit, node handle, quality of service, connect type, port event handler, user information". In Aldred's system, the sending application must provide the support system with this information in channel creation; but *it does not establish the channel itself.* Applicants also note that in creating the channel between two programs, i.e., the launched and launching programs, a target application handle and a node handle are required by the support system. The Examiner speculates (erroneously) that "parameters passed in Aldred's launch_app function would be the channel characteristics needed for the launching and launched applications to communicate". However, the Examiner has not cited any portion of the art to support such a suggestion, and as noted above, a target application handle and a node handle are required for the channel creation, and in 11:27-39, are not available until *after* the application has been launched. Furthermore, the node handle is specified by the return data, which is returned **after the application has registered** with the support system (11:29-31, 11:36-39).

Furthermore, contrary to the Examiner's assertion in Response to Arguments, the limited use handle does not disclose "that a channel has been already established between the applications". In fact, Aldred defines how the handle is implemented in 36:45-48: "This function [launch_app] is used to ask the system to start another program instance. IF the new application is started successfully then its instance handle is inserted in the target_application and returned to the calling application". Aldred has already defined the target_application as a pointer used by the system. Therefore, Aldred's system changing the value of a particular pointer, where that changing of value is communicated within the support system and not directly between the two programs, does not teach that a channel has been already established between the applications. And, as noted above, the newly launched application is not "allowed to communicate with the system" without registration. Furthermore, this communication with the system is required for the system to create a channel as in the API call function add_channel or create_channel (col. 29).

Moreover, modifying Aldred's system to include passing an event port number and a command port number to an application as part of launching that application would change the principle of operation of Aldred's system. Aldred's system relies upon applications registering and utilizing both the call managers and the support system software via Aldred's API to properly initiate and configure communications between applications. Bypassing this system to send event port numbers and command port numbers to applications, as part of launching those applications, would bypass Aldred's sharing set concept, which is essential to the operation of his system, and thus change Aldred's principle of operation. As discussed in § 2143.01 of the M.P.E.P, a rejection based on a modification that changes the principle of operation of a reference is improper.

Additionally, the Examiner characterizes Aldred's registration to the support system as "the registration by the launched application is merely to allow other applications to know that it has been launched". Furthermore, the Examiner has stated that nowhere does Aldred declare that ports are only configured after an application has registered with the support system. Apparently the Examiner has overlooked the fact that Aldred specifically discloses that: "In order for an application instance to be allowed to communicate with the system, it must identify itself by issuing an register_app call. This call must be issued prior to any other calls from this instance, otherwise the calls will fail." Additionally, Aldred teaches that "it is up to the launched application to use an register_app to fully identify itself to the system" (35:47-67). Clearly, Aldred's registration allows the newly launched application to interact with the system and is not simply limited to "allow other applications to know that it has been launched".

In another instance, the Examiner speculates, "a reason for registration might be to prevent already launched programs from being launched again" citing 42:55-58: "The Call Manager makes a note of the handle of the launched program, so that when the corresponding Register_app event occurs it doesn't try to share it again". However, Applicants note that, just below this citation, in 42:65-68, "the Call Manager described here never resolves an incoming share request by sharing with an application that is already running—*a new instance is always launched*" (emphasis added). Clearly, the registration does not "prevent already launched programs from being launched again" based on this text. The Examiner continues his unsupported speculation, based on previously assumed speculation, in Response to Arguments page 4: "The possibility that a channel has been already established between the applications and that the registration by the launched application is merely to allow other applications know that it has been launched imply that the launch_app function parameters are involved in providing the necessary information to the launched applications." As argued above, the channel has not already been established due to the necessity of the node_handle and application_handle which are received after the program has launched, and, the registration process is not merely to allow the other applications to know the application has been launched. Thus, the Examiner's implication is unfounded.

As argued above, the rejection of claim 1 is not supported by Aldred. Furthermore, the combination with Raynak does not overcome the above-noted deficiencies of Aldred. Thus, for at least the reasons provided above, Applicants submit that neither Aldred nor Raynak, either singly or in combination, discloses the features and limitations of claim 1.

Thus, for at least the reasons presented above, the rejection of claim 1 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks also apply to claims 12 and 20.

Regarding claim 5, Aldred in view of Simonoff does not teach or suggest passing a function reference value through the command port connection. The Examiner cites column 24, lines 52-61 of Aldred. However, this portion of Aldred is referring to how applications can make asynchronous calls to Aldred's support system software API by including a reference identifier allowing the application to issue command to obtain the status of, or to cancel, an asynchronous API call. The cited passage does not describe passing a function reference value through a command port connection, as suggested by the Examiner. The reference identifier mentioned in the cited passage is not a function reference value that is sent through a command port connection. Instead, it is merely an identifier to allow a calling application to check on the status of an asynchronous API call.

In the Response to Arguments, the Examiner asserts that Aldred's reference identifier passes through the command port. However, in Aldred, the status of an API call is handled between the application issuing the call and the support system, and not between the launching and launched applications. The support system is only inherent in Aldred's system, and therefore, when communication occurs between only an application and the support system, Aldred's command port does not equate to that of the instant application. The Examiner does not refer to Simonoff in the rejection of claim 5. Aldred does not disclose passing a function reference value *through a command port*. Nor does Simonoff overcome Aldred's failure to teach or suggest passing a function reference value through a command port connection. Thus, the rejection of claim 5 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks apply to claim 16.

Regarding claim 6, Aldred and Raynak, in further view of Simonoff, does not teach or suggest passing a function parameter through the command port connection. The Examiner cites column 24, lines 39-42 of Aldred. However, this passage of Aldred describes how Aldred's system handles an application's request for a service. Specifically, Aldred teaches that an application requests a service and supplies the appropriate parameters. Another application, supplying the service, is made aware of the request through an "unsolicited event which appears as an indication" to the second application. The response from the second application is routing back to the requesting application as a "confirm event." The cited passage does not mention passing a function parameter through a command port connection, but instead teaches the use of unsolicited events to request a service and to deliver responses. Elsewhere (column 7, lines 44-62) Aldred teaches three different types of port connections: event, command and null ports. The passage cited by the Examiner in the rejection of claim 6 clearly refers to issuing events via event ports. The cited passages does not mention any command port connection.

In the Response to Arguments, the Examiner reasserts that the parameters are submitted through the command port because "command ports allow the application to drive the receipt or supply of *data* to the port". The Examiner is clearly redefining parameters of data using hindsight speculation which is improper. Furthermore, the Examiner cites columns 35 and 36 and asserts "functions are submitted through the command port". The cited text does not disclose command ports at all; in fact, columns 35 and 36 repeatedly disclose events and event handlers associated with the functions that would be communicated through the event port. The cited section in column 24 clearly defines the process associated with the "appropriate parameters" through events. Events are passed along the event port. The Examiner speculates that parameters are passed along the command port, but nowhere provides evidence or an example to support this. Thus, the rejection of claim 6 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks also apply to claim 17.

Applicant also asserts that numerous ones of the dependent claims recite further distinctions over the cited art. However, since the rejection has been shown to be unsupported for the independent claims, a further discussion of the dependent claims is not necessary at this time.

CONCLUSION

Applicants submit the application is in condition for allowance, and prompt notice to that effect is respectfully requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5681-78901/RCK.

Also enclosed herewith are the following items:

☒ Return Receipt Postcard

Respectfully submitted,

Robert C. Kowert
Reg. No. 39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800

Date: November 18, 2005